

AMENDMENTS TO THE CLAIMS

1. (Currently Amended) A method of providing filtered unified logging, the method comprising:

receiving a message relating to unified logging and tracing, the message having a predefined severity, the message further including a trace-tracing message and a log-logging message, wherein the trace-tracing message is associated with a location, the location representing a delimited code area including one or more of a code package, and a code class, wherein the log-logging message associated with a category, the category representing a code problem area including one or more of a database problem, and a security audit problem;

dropping the message if the message predefined severity does not reach a threshold severity;

applying one or more filters to perform a filter evaluation of the message if the message predefined severity reaches the threshold severity, and publishing the message if it reaches the threshold severity and passes the filter evaluation, wherein the filter evaluation is to determine whether the message is to be restricted, screened out, or published with an altered behavior; and

sending the message to a destination if the message is not filtered out fails the filter evaluation.
2. (Currently Amended) The method of claim 1, wherein each of the message predefined severity and the threshold severity have having a severity selected from a group comprising one or more of debug, path, info, warning, error, fatal, and none.

3. (Currently Amended) The method of claim 1, further including comparing the message-predefined severity with the threshold severity.
4. (Currently Amended) The method of claim 1, wherein the predefined severity information is and the threshold severity are inherited by a child message of the message, if a predefined severity of the child message reaches the threshold severity and child message passes the filter evaluation.
5. (Cancelled)
6. (Cancelled)
7. (Cancelled)
8. (Original) The method of claim 1, wherein the method is implemented utilizing Java.
9. (Currently Amended) The method of claim 1, wherein the message is sent by an application at a server.
10. (Cancelled)
11. (Currently Amended) A filtered unified logging system, system comprising:
a server comprising a processor and a storage medium coupled with the processor,
the storage medium having instructions to execute a unified logging and
tracing system, wherein the instructions are executed by the processor, the unified logging and tracing system having
one or more log controllers to represent source data, each of the log controllers

receiving messages a message, the messages include having a predefined severity, the message further including a trace tracing message and a log logging message, the trace tracing message is associated with a location, the location representing a delimited code area including one or more of a code package, and a code class, the log logging message is associated with a category, the category representing a code problem area including one or more of a database problem, and a security audit problem, the one or more controllers to drop the message if the predefined severity does not reach a threshold severity;

one or more filters coupled to the log controllers to perform a filter evaluation of filter the received messages message if the predefined severity reaches the threshold severity, the one or more filters to publish the message if it reaches the threshold severity and passes the filter evaluation, wherein the filter evaluation is to determine whether the message is to be restricted, screened out, or published with an altered behavior, the one or more filters to send the message to a destination if the message fails the filter evaluation;

one or more logs to represent one or more corresponding destinations for the received messages; and

one or more filters coupled to the logs to filter the messages prior to publication by the logs;

12. (Currently Amended) The filtered unified logging system of claim 11, wherein the system is part of Java 2 Enterprise Edition (J2EE) engine.

13. (Currently Amended) The ~~filtered-unified-logging system~~ of claim 11, further including a formatter coupled to ~~each of the logs~~ ~~log controllers~~ to determine a format of the ~~received~~ message prior to publication.
14. (Currently Amended) The ~~filtered-unified-logging system~~ of claim 13, wherein the formatter includes one or more subclasses or modules selected from a group comprising a list formatter, a trace formatter, and an Extensible Markup Language (XML) formatter.
15. (Cancelled)
16. (Currently Amended) The ~~filtered-unified-logging system~~ of claim 11, wherein each of the logs include one or more subclasses or modules selected from a group comprising a stream log, a file log, and a console log.
17. (Cancelled)
18. (Cancelled)
19. (Cancelled)
20. (Currently Amended) The ~~filtered-unified-logging system~~ of claim 19, wherein the ~~predefined severity and threshold severity information~~ includes at least one a severity selected from a list comprising debug, path, info, warning, error, fatal, and none.
21. (Currently Amended) The ~~filtered-unified-logging system~~ of claim 19, wherein the ~~predefined severity information is and the threshold severity are inherited by a child message of the message, if a predefined severity of the child message~~

reaches the threshold severity and child message passes the filter evaluation.

22. (Cancelled)

23. (Cancelled)

24. (Currently Amended) The ~~filtered-unified-logging~~ system of claim 11, wherein the system is implemented using Java.

25. (Currently Amended) A ~~computer program for providing machine-readable storage medium comprising instructions~~~~filtered-unified-logging~~, the computer program which, when executed, cause a machine to ~~comprising~~:

~~a machine-readable medium that provides instructions that, if executed by a machine, will cause the machine to perform operations including:~~

~~receiving a message relating to unified logging and tracing, the message having a predefined severity, the message further including a trace tracing message and a log-logging message, wherein the trace tracing message is associated with a location, the location representing a delimited code area including one or more of a code package, and a code class, wherein the log-logging message associated with a category, the category representing a code problem area including one or more of a database problem, and a security audit problem;~~

~~dropping the message if the message predefined severity does not reach a threshold~~

severity;

applying one or more filters to perform a filter evaluation of the message

if the message-predefined severity reaches the threshold severity,

and publish the message if it reaches the threshold severity and

passes the filter evaluation, wherein the filter evaluation is to

determine whether the message is to be restricted, screened out, or

published with an altered behavior; and

sending the message to a destination if the message fails the filter

evaluation is not filtered out.

26. (Currently Amended) The computer program-machine-readable storage medium of claim 25, wherein the message-predefined severity and the threshold severity have include a severity selected from a group comprising debug, path, info, warning, error, fatal, and none.
27. (Currently Amended) The machine-readable storage medium computer program of claim 25, wherein the operations further include instructions which, when executed, further cause the machine to comparing compare the message severity with the threshold severity.
28. (Currently Amended) The computer program of claim 25, wherein the predefined severity information is and the threshold severity are inherited by a child message of the message, if a predefined severity of the child message reaches the threshold severity and child message passes the filter evaluation.
29. (Cancelled)

30. (Cancelled)
31. (Currently Amended) The computer program of claim ~~30~~²⁵, wherein the publication is via an inherited log.
32. (Cancelled)
33. (Currently Amended) The computer program of claim 25, wherein the message is sent by an application ~~at a server~~.
34. (Cancelled)